AFRL-SR-AR-TR-04-

0579

REPORT DOCUMENTATION PAGE

1. REPORT DATE *(DD-MM-YYYY)*
2 – November- 2004

2. REPORT TYPE
Final Performance Report

3. DATES COVERED *(From - To)*
15-May-2001 to 15-May-2004

4. TITLE AND SUBTITLE
Multimodality Image Fusion for 3D Model Building with Applications

5a. CONTRACT NUMBER

5b. GRANT NUMBER
F49620-01-1-0367

5c. PROGRAM ELEMENT NUMBER

6. AUTHOR(S)
Farag, Aly A. (Project Director)

5d. PROJECT NUMBER

5e. TASK NUMBER

5f. WORK UNIT NUMBER

7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES)
Computer Vision & Image Processing Laboratory
Rm 412 Lutz Hall
University of Louisville
Louisville, KY 40292

8. PERFORMING ORGANIZATION REPORT NUMBER

9. SPONSORING / MONITORING AGENCY NAME(S) AND ADDRESS(ES)
DFAS-OM/FP
P.O. Box 7020
Bellevue, NE 68005-1920
(800)330-8168

NL

10. SPONSOR/MONITOR'S ACRONYM(S)
F25700 (code)

11. SPONSOR/MONITOR'S REPORT NUMBER(S)

12. DISTRIBUTION / AVAILABILITY STATEMENT
Approve for Public Release:Distribution Unlimited

13. SUPPLEMENTARY NOTES

14. ABSTRACT

This project deals with multimodality image fusion for the purpose of construction robust 3D model of a target area. During 2002, some classification algorithms and basic data fusion techniques have been performed. Graduate student and a research scientist have been hired to assist in carrying out the research plan.

15. SUBJECT TERMS

16. SECURITY CLASSIFICATION OF:

a. REPORT	b. ABSTRACT	c. THIS PAGE

17. LIMITATION OF ABSTRACT

18. NUMBER OF PAGES
6

19a. NAME OF RESPONSIBLE PERSON
Aly A. Farag

19b. TELEPHONE NUMBER *(include area code)*
(502)852-7510

MULTIMODALITY IMAGE FUSION FOR 3-D MODEL BUILDING WITH APPLICATIONS

Professor Aly A. Farag

CVIP Laboratory
University of Louisville, Louisville, KY 40292
E-mail: farag@cairo.louisville.edu
Phone: 502-852-7510; Fax: 502-852-1580

Overall Project Plan

In this investigation, we propose a methodology for 3-D model building by the fusion of multimodality data provided from space-borne and/or air-borne sensors. Figure 1 illustrates a conceptual framework for multisensory data fusion for 3-D model building. A 3-D model of a target area can be built using different data types, e.g., Landsat MSS data, AVIRIS hyperspectral data, range data, and/or elevation (DEM) data.

Data fusion and integration can be performed either at the data level or at the decision level. Multispectral and hyperspectral data sets can be classified locally (e.g., using the fuzzy c-mean classifier), then decision fusion are used to fuse the local decision classes, or data fusion techniques can be used to fuse the data sets into one data set for classification. Probabilistic and evidential methods for data fusion will be investigated in this study.

Topographic data, from range scanners (ALTM) and/or radar (DTED) can be integrated, after registration, with the classification results of the multispectral and hyperspectral data in order to produce the final 3-D model of the sensed target area.

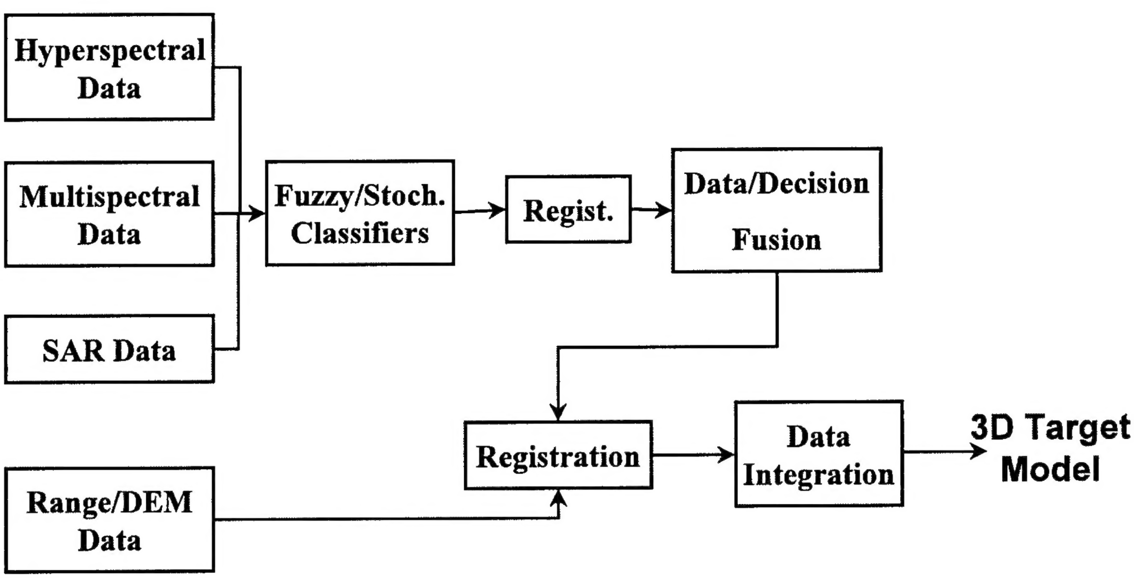

Figure 1: *Block diagram of the proposed classification and data fusion methodology.*

20041119 017

Efforts Done During the Grant Period

The following research achievements summarize the efforts done during the grant period.

- **Efforts during 2001**

  - **Data collection**

We created a data-fusion group on the SGI ONYX-II supercomputer at the CVIP Lab. That group has access to over 200GB of remote sensing data from various sources. We also installed a one-seat license of the ENVI software package to be used for data display and basic image processing tasks. Standard statistical and numerical analysis packages are readily available at the CVIP Lab and are being used in this project.

  - **Fuzzy classification for hyperspectral data**

We are pursuing the development of the Fuzzy C-Means (M-FCM) algorithm that was introduced in 2000 [1]-[3]. Our goal is to make the M-FCM unsupervised in terms of the number of classes, and the parameters per class.

  - **Fusion of range and stereo data**

Range data generated by a laser scanner at the CVIP Lab (Cyberware 3030RGB) are generated for various objects (including human faces). At the same time, calibrated sequence of 2D images is generated, for the same objects. Shape extraction methods (e.g., stereo and space carving [4]) are used to generate 3D objects, which will be calibrated/integrated using the laser scanned models. Purpose of this experiment is to generate some ground truth data to test the basic data fusion and integration models to be developed in this research.

- **Efforts during 2002**

  - **Updating the available programs and testing them with new data sets**

We updated the programs that implemented by the lab staff worked in similar projects at the lab. These algorithms include the Bayes' classifier, K-Nearest neighbors' classifier, and Fuzzy logic classifier. The class conditional probability is implemented using parametric methods "e.g. Gaussian model" and parametric methods "e.g. Parzen window" for the Bayes' classifier verification. These algorithms are tested using two Lansat data sets, the golden bay area of San Francisco and an agricultural area. The classification accuracy of these algorithms is roughly about 92 % for the golden bay area and about 70% for the agricultural area. Detailed results can be found in [1].

  - **Statistical, fuzzy logic classifiers and data fusion algorithms**

We implemented some stochastic (statistical) algorithms that depend on modeling the classification process as stochastic processes. These models are represented as Gibbs-Markov random fields the Iterative Conditional Modes (ICM) algorithm is used for their implementation. These methods enhance significantly the classification accuracy of the previously implemented algorithms. For the golden bay area, the accuracy reaches 97.5% where it reaches 90% for the agricultural area. Also, a fuzzy logic classifier is implemented, [5].

A simple data fusion algorithm emerged from the well known Bayes' rule is implemented, [5]. This algorithm is applied to fuse the output of two classifier outputs. The results show that the fusing algorithm enhances the results, but still more works is required in this direction.

- **Feature extraction**

Some effort is done for feature extraction of Landsat data. These features depend mainly on first and second order histogram features. But still this effort is under developing and needs some enhancement to provide good characterizing and representative features.

- **Image registration and fusion**

In addition to the tasks scheduled for the period of this report, there was a considerable effort done for image registration and fusion. A method for image registration using parallel genetic algorithms is proposed and implemented. Some experiments with Landsat data sets are done to test the effectiveness of the method. The results show that the method has the ability to register two data sets effectively and within considerable proceeding time, [6].
A general purpose platform for 3-D model registration and fusion is founded in the lab. The main component of this platform is 3-D laser scanner which is available in the lab. With this platform, 3-D models can be registered and fused. Up to now, the platform is tested using 3-D generated by Computer Vision algorithms, like stereo based and space carving methods, [7]. It is planned to use platform in registering and fusion of 3-D models extracted from spectral data sets.

o **Efforts during 2003**

- **High Dimensional Density Estimation Algorithms**

We designed and implemented a density estimation algorithm appropriate for higher dimensional data sets. This algorithm uses the Support Vector Machines (SVM) method for density estimation. The algorithm is applied for various data sets, either synthetic or real data, in various dimensions. The results showed that the algorithm is suitable and uséful for density estimation in higher dimensions. The algorithm is used in Bayes setup for the estimation of class probabilities for the various classes in multispectral data sets. The proposed algorithm is compared with previously designed algorithms and the results showed the superiority of the proposed algorithms over other algorithms. Detailed results can be found in [8, 12].

- **Statistical Multi-Stages Classifier for Multispectral Data**

We designed and implemented some stochastic algorithms that consist of two stages. The first stage uses the SVM to implement the class conditional probabilities in a Bayes classification setup. The second stage models the classification process as stochastic processes. These models are represented as Gibbs-Markov random fields and the Iterative Conditional Modes (ICM) algorithm is used for their implementation. The results show that the two stages setup enhances significantly the classification accuracy of the previously implemented algorithms. For the golden bay area, the accuracy reaches 98.5% where it reaches 90% for the agricultural area. Detailed results can be found in [9, 11].

- **Unsupervised Statistical Classification**

It is not the usual case that there are some training samples that can be used to estimate the parameters of a classifier. So, we designed and implemented a classification algorithm that works in a complete unsupervised setup. The algorithm is unsupervised in terms of the number of classes defined in a data set. Also, the algorithm is unsupervised in the sense that it does not need training samples for each class.
The algorithm considers a data set as a mixture of densities and it uses the well-known EM algorithm with some modifications. The algorithm estimates the number of classes, the parameters of each class

density as long as the proportions of each class in the data set. We applied the algorithms for various data sets, either synthetic or real data. The results show that the algorithm performs well and comparable to supervised algorithms. Detailed results can be found in [10].

- **Efforts during 2004**

  - **Mean Field Theory for Density Estimation using Support Vector Machines**

A learning algorithm is developed for learning of SVM as a density estimator. This algorithm depends on the Mean Field Theory which compresses the complicated statistical interactions of random variables into a simple distribution. The algorithm is used to enhance the performance of the SVM density estimation in both time and accuracy consideration. The results [13] on both synthetic and real multispectral data illustrate that the proposed algorithm is highly fast with respect to the traditional formulation of the SVM density estimator. Also, the results show an enhancement of the performance of the density estimation process. The real multispectral data classification is used in the performance evaluation as an application for density estimation.

  - **A Unified Framework for MAP Estimation in Remote Sensing Image Segmentation**

This is an ongoing work which aims at setting a framework for applying the MAP segmentation principle on remote sensing data [14]. An algorithm is proposed for each part of the MAP principle as well as an iterative setup for the whole system. The Mean Field based SVM density estimation is used for the class conditional probabilities estimation and MRF is used for the modeling of the high level process of the MAP principle. A new algorithm which is simple and fast for the estimation of the MRF parameters is proposed. An iterative setup is used to maximize the MAP estimate so that more enhanced segmented image can be obtained.

Interactions

During the time period considered in this report, the following seminars and talks took place:

I. "Remote Sensing Data Analysis and Data Fusion: Some Concepts and Algorithms Overview,"
By: Refaat M Mohamed, CVIP Lab, Wed. March 27-02.

II. "Multi-Spectral Data Classification: 'Primary Results',"
By: Refaat M Mohamed, at the CVIP Lab., Wed April 3-02.

III. "Experiments in Multimodality Image Classification and Data Fusion,"
By: Refaat M Mohamed, IF'02 Conference, Annapolis, MD, Monday Jul 8-02.

IV. "A General-Purpose Platform for 3-D Reconstruction from Sequence of Images,"
By: Refaat M Mohamed, IF'02 Conference, Annapolis, MD, Monday Jul 8-02.

V. "Image Registration in Multispectral Data Sets,"
By: Hani Mahdi, at CVIP Lab., Tuesday Jul 30-02.

VI. "Developments in Remote Sensing Data Classification and Fusion"
By: Aly A Farag, at the AFOSR meeting, Washington DC, Monday Aug 12-02.

VII. "Recent Results in Bayesian Classification of Multispectral Data,"

By: Refaat M Mohamed, ASPRS-Midsouth 2002, Murray, Kentucky, Thursday Oct-24, 2002.

VIII. "Application of Support Vector Machines in Density Estimation,"
A poster by: Refaat M Mohamed, ASPRS-Midsouth 2002, Murray, Kentucky, Friday Oct-25, 2002.

IX. "Classification of Multispectral Data Using Support Vector Machines Approach for Density Estimation,"
By: Refaat M Mohamed, International Conference on Intelligent Engineering System INES 2003, Assiut, Egypt, Tuesday March-6, 2003.

X. "Unsupervised Density Estimation Using EM Algorithm,"
By: Refaat M Mohamed, at CVIP Lab., Wednesday April-3, 2003.

XI. "Application of Support Vector Machines in Density Estimation,"
A poster by: Refaat M Mohamed, EPSCoR 2003, May-12, 2003.

XII. "Two Sequential Stages Classifier for Multispectral Data,"
By: Refaat M Mohamed, at IEEE conference on Computer Vision and Pattern Recognition (CVPR) Workshop on Intelligent Learning 2003, Sunday June-22, 2003.

XIII. "Mean Field Theory for Density Estimation using Support Vector Machines,"
By: Aly A Farag, at IF'04 Conference, Stockholm, Sweden, July 29-04.

References

1. M. G-H Mostafa, T. Perkins and A. Farag, "A Two-Step Fuzzy-Bayesian Classification for High Dimensional Data", Proc. International Conference on Pattern Recognition (ICPR'2000), Barcelona, Spain, volume 3, pages 421-424, September 2000.
2. M. G-H. Mostafa, A. A. Farag and E. A. Essock, "Multimodality Image registration and Fusion using Neural Network," Fusion2000, Paris, France, Vol. 2, pages WeD3-3 -- WeD3- 9, July 2000.
3. Aly Farag and Timothy Perkins, "Data Fusion in Aerial Imaging Applications," TR-CVIP, December 2000.
4. Aly Farag and Abdoulah Sangare, "Space Carving for 3-D Object Modeling," TR-CVIP, December 2000.
5. Aly A. Farag, Refaat M. Mohamed and Hani Mahdi, "Experiments in Multimodality Image Classification and Data Fusion," Proceedings of the Fifth International Conference on Information Fusion, July 7-13-02, Annapolis, MD, pp. 299-308.
6. Hani Mahdi and Aly A. Farag, "Image Registration in Multispectral Data Sets," to be appeared in the Proceedings of the International Conference on Image Processing (ICIP 2002), Sept 22-25-02, Rochester, New York.
7. Ahmed H. Eid, Sherif S. Rashad, and Aly A. Farag, "A General-Purpose Platform for 3-D Reconstruction from Sequence of Images," Proceedings of the Fifth International Conference on Information Fusion, July 7-13-02, Annapolis, MD, pp. 425-431.

8. Refaat M. Mohamed, Aly A. Farag," Classification of Multispectral Data Using Support Vector Machines Approach for Density Estimation," Proceedings of the International Conference on Intelligent Engineering System INES 2003, Assiut, Egypt, March 6-8, 2003.

9. Refaat M Mohamed and Aly A. Farag, "Two Sequential Stages Classifier for Multispectral Data," Proceedings of the International Conference on Computer Vision and Pattern Recognition (CVPR) 2003 workshop on Intelligent Learning, Madison, WS, June 16-22, 2003.

10. Refaat M Mohamed and Aly A. Farag, "A New Unsupervised Approach for the Classification of Multispectral Data," Proceedings of the Sixth International Conference on Information Fusion, July 8-11-03, Cairns, Queensland, Australia, , pp. 951-959.

11. Ayman S El-Baz and Aly A. Farag, "Parameter Estimation in Gibbs-Markov Image Models," Proceedings of the Sixth International Conference on Information Fusion, July 8-11-03, Cairns, Queensland, Australia, pp. 934-943.

12. Refaat M Mohamed and Aly A. Farag, "Parameter Estimation for Bayesian Classification of Multispectral Data," Proceedings of the Seventh International Conference on Knowledge-Based Intelligent Information & Engineering Systems, University of Oxford, United Kingdom, September 4-5 2003.

13. Aly A Farag and Refaat M Mohamed, "Mean Field Theory for Density Estimation using Support Vector Machines," Proceedings of the Seventh International Conference on Information Fusion, June 29, July 2,04, Stockholm, Sweden , pp. 951-959.

14. Aly A Farag, Refaat M Mohamed and Ayman El-Baz, "A Unified Framework for MAP Estimation in Remote Sensing Image Segmentation," to be submitted to the IEEE Transactions on Geoscience and Remote Sensing.

Appendix: Software

The following listings summarize some of the code used in the development of the research efforts discussed before. All the listings are coded in MATLAB 6 or higher.

- **KNN Classification**

```
clear all
format long
load EvalData %Contanins Test data "Z" and the corresponding desired "D"
ZTest=Normalize(ZTest);             % choose side length
load TrainData %Contains ProtoTypes "Training Data" and ProtoDisered "Training Data Desire"
Z=Normalize(Z);

for i=1:length(DTest)
 i
  Distances=dist(ZTest(:,i)',Z);
    [val indx]=min(Distances);
  for j=1:15
    [val indx]=min(Distances);
    m(j)=D(indx);
    Distances(indx)=max(Distances);
  end
  for     k=0:8
    cand(k+1)=length(find(m==k));
  end
  [val indx]=max(cand);
  d(i)=indx-1;
end
OverAllAccuracy=100*length(find((DTest-d)==0))/(length(DTest))
fprintf ('Overall Accuracy = %f\n',length(find((DTest-d)==0))/(length(DTest)))

fprintf('Class\tTotal\tClass0\tClass1\tClass2\tClass3\tClass4\tClass5\tClass6\tClass7\tClass8\tAccuracy\n');
ConfKNN(1,:)=classmatrx(DTest,d,9,0);
ConfKNN(2,:)=classmatrx(DTest,d,9,1);
ConfKNN(3,:)=classmatrx(DTest,d,9,2);
ConfKNN(4,:)=classmatrx(DTest,d,9,3);
ConfKNN(5,:)=classmatrx(DTest,d,9,4);
ConfKNN(6,:)=classmatrx(DTest,d,9,5);
ConfKNN(7,:)=classmatrx(DTest,d,9,6);
ConfKNN(8,:)=classmatrx(DTest,d,9,7);
ConfKNN(9,:)=classmatrx(DTest,d,9,8);
```

- **Parzen-Window based Classification**

```
clear all
format long
Priors=[0.23773677392248  0.32810475823676  0.29603305206400  0.06732957529498         0.02801022373166
0.00227583067820  0.02167291061237  0.00497181471237 0.01386506074717]';
load TrainData
D0Length=length(find(D==0));  % No Of Points in the space
D1Length=length(find(D==1));
D2Length=length(find(D==2));
D3Length=length(find(D==3));
D4Length=length(find(D==4));
D5Length=length(find(D==5));
```

```
D6Length=length(find(D==6));
D7Length=length(find(D==7));
D8Length=length(find(D==8));
ClassLengths=[D0Length D1Length      D2Length      D3Length      D4Length      D5Length
        D6Length      D7Length      D8Length];
Z=Normalize(Z);
load EvalData                  %Y = X from original test data for initial testing purposes
ZTest=Normalize(ZTest);        % choose side length
Radius = 0.1;                          % seems to match Gaussian Kernel results well

for i = 1:size(ZTest,2)    %Test all the test data ; remove this loop in the single point function
  i
  Distances=dist(ZTest(:,i)',Z);
  [Val Indxs]=find(Distances<Radius);
  Classes=D(Indxs);
  if (length(Classes)>0)
          D0=length(find(Classes==0));  % No Of Points in the Volume sphere
        D1=length(find(Classes==1));
                D2=length(find(Classes==2));
                D3=length(find(Classes==3));
                D4=length(find(Classes==4));
                D5=length(find(Classes==5));
                D6=length(find(Classes==6));
                D7=length(find(Classes==7));
                D8=length(find(Classes==8));
                Lengths=[D0    D1    D2    D3    D4    D5    D6    D7    D8];
        Prop=Lengths./ClassLengths;
    p = Prop'/(sum(Prop));                %so you can see side-by-side with WorkSpace browser
  else
    p = zeros(9,1);               %so you can see side-by-side with WorkSpace browser
  end
  Pwx=p.*Priors;
  [indx val]=find(Pwx==max(Pwx));
  if (length(indx)==1)
    d(i)=indx-1;
  else
    d(i)=10;
  end
end

fprintf ('Overall Accuracy = %f\n',length(find((DTest-d)==0))/(length(DTest)))

fprintf('Class\tTotal\tClass0\tClass1\tClass2\tClass3\tClass4\tClass5\tClass6\tClass7\tClass8\tAccuracy\n');
ConfBayes456(1,:)=classmatrx(DTest,d,9,0);
ConfBayes456(2,:)=classmatrx(DTest,d,9,1);
ConfBayes456(3,:)=classmatrx(DTest,d,9,2);
ConfBayes456(4,:)=classmatrx(DTest,d,9,3);
ConfBayes456(5,:)=classmatrx(DTest,d,9,4);
ConfBayes456(6,:)=classmatrx(DTest,d,9,5);
ConfBayes456(7,:)=classmatrx(DTest,d,9,6);
ConfBayes456(8,:)=classmatrx(DTest,d,9,7);
ConfBayes456(9,:)=classmatrx(DTest,d,9,8);
```

- **Density Estimation Using SVM**

```
clear all
close all
format long
```

```
load data1D X n
X=X';
Sigma=0.9;
N=n;
[XSorted        I]=sort(X);
Y=[];
Epsilon=[];
for i=1:N
  n=length(find(XSorted<=XSorted(i)));
  Y=[Y;n/N];
  Epsilon=[Epsilon;sqrt((1/N)*Y(i)*(1-Y(i)))];
end

Tolerance=1e-5;
n=size(XSorted,1);
for i=1:n
  for j=1:n
    h(i,j)=Kernel1( XSorted(i,:),XSorted(j,:),'density',Sigma);
    q(i,j)=Kernel1( XSorted(i,:),XSorted(j,:),'distribution',Sigma);
  end
end
H=2*h;%H=H+1e-10*H;
f=zeros(1,n);
B=[Epsilon+Y];
%B=[Epsilon+Y;Epsilon-Y];
A=[q];
%A=[q;-q];
Aeq=[ones(1,n)];
beq=1;
Lower=zeros(n,1);
Upper=inf*ones(n,1);
[Beta    b]=quadprog(H,f,A,B,Aeq,beq,Lower,Upper);
SV=find( (abs(Beta)>Tolerance));
nSV=length(SV)
%pause

%X1=[-2+XSorted(1):0.01:XSorted(length(XSorted))+2]';
X1=[min(X)-3:0.1:max(X)+3]';      %For 1D Gaussian
A=sqrt(2*pi)*Sigma;
for i=1:size(X1,1)
  Tempy=0;
  Tempfx=0;
  for j=1:nSV
    Tempy=Tempy+Beta(SV(j))*inv(A)*Kernel1( X1(i,:),X(SV(j),:),'distribution',Sigma);
    Tempfx=Tempfx+Beta(SV(j))*inv(A)*Kernel1( X1(i,:),X(SV(j),:),'density',Sigma);
  end
  FX(i)=Tempy;
  fx(i,1)=Tempfx;
end
Ref=normpdf(X1);
h2=normcdf(X1);

fxModifies=abs(max(Ref)*fx/max(fx));
KBD01=sum(fxModifies.*log(fxModifies./Ref))       %Kullback-Leibler distance
KBD10=sum(Ref.*log(Ref./fxModifies))       %Kullback-Leibler distance
figure
plot(X1,Ref,'k');
```

```
hold on
plot(X1,fxModifies,'.k')
legend('True', 'Estimated'); axis([-4  4  0  0.44]);grid on
```

- **Stochastic Modeling and SVM for Multispectral Data Classification**

```
%	This module implements the Stochastic modeling approach for Multispectral data
%	Classification. The inputs are:
%	1-	Y		:	Data observations in the form of m*n where m is the observation
%				dimension, n is the number of observations.
%	2- D		:	The ground truth data. (matrix if it is for the whole image).
%	3- Image0	:	The Initil guess to start relaxation by ICM.
%	4- Mu		:	The collected Mean vectors (m*C), C is the # of classes.
%	5- Segma	:	The collected Mean vectors ((m*C)^2), C is the # of classes.
%	The associated functions are : 1) Besage	2) Normal	3)Classmatrx 4)writePPM
clear all
close all
format long
%global Image0  Beta
load C:\Refaat\MultiSpect\Data\Agricultural\AllObservations
load InitialSVMGuess

NoOfClasses=max(max(D));
L=169;
W=169;
imshow(D*50,[0 255]); Title('Reference');
figure
imshow(Image0*50,[0	255]); Title('Initial Guess');
[Indx1 Indx2]=find((Image0-D)==0);
length(Indx1)/(169*169)

InitialImage=Image0;

Image=zeros(L+1,W+1);
Image(1,2:(L+1))=Image0(L,:);
Image(L+2,2:(L+1))=Image0(1,:);
Image(2:(L+1),1)=Image0(:,W);
Image(2:(L+1),W+2)=Image0(:,1);
Image(2:(L+1),2:(W+1))=Image0;
Image0=Image;

NoOfIterations=5;
Beta=-6;
for N=1:NoOfIterations
  for i=2:(L+1)
    for j=2:(W+1)
      for C=1:NoOfClasses
        p(C)=P(C,(i-2)*W+(j-1))*Besage(i,j,C);
      end		% C-Statementw
      [Value Indx]=max(p);
      Image(i,j)=Indx;
    end  % J-Statement
  end	% I-Statement
  Image0=Image;
        figure
        imshow(Image*50,[0	255]);Title(strcat('After	',num2str(N)));
  [Indx1 Indx2]=find((Image(2:L+1,2:W+1)-D)==0);
  length(Indx1)/(169*169)
```

```
  N
%        pause
end      % K-Statement
BesageFinalImage=Image(2:(L+1),2:(W+1));
fprintf('Class\tTotal\tClass0\tClass1\tClass2\tClass3\tClass4\tClass5\tClass6\tClass7\tClass8\tAccuracy\n');
classmatrx(BesageFinalImage,D,9,1);
classmatrx(BesageFinalImage,D,9,2);
classmatrx(BesageFinalImage,D,9,3);
classmatrx(BesageFinalImage,D,9,4);
classmatrx(BesageFinalImage,D,9,5);
classmatrx(BesageFinalImage,D,9,6);
classmatrx(BesageFinalImage,D,9,7);
classmatrx(BesageFinalImage,D,9,8);
classmatrx(BesageFinalImage,D,9,9);

ColorCode=[   255    0    0        0        0        255    255    255    128
        128
                     255    0    0        255    255    0        0        255
        128    128
       255    0    255    0        255    0        255    0        128    255
       ];
for i=0:(L-1)
        for j=1:W
    R( (i*3+1):(i*3+3),j)=ColorCode(:,D(j,i+1)+1);
  end
end
writeppm(R,'Original1.ppm',L,W);

for i=0:(L-1)
  for j=1:W
    R( (i*3+1):(i*3+3),j)=ColorCode(:,BesageFinalImage(j,i+1)+1);
  end
end

writeppm(R,'FinalBesage1.ppm',L,W);
```

- **SVM Density Estimation Using Mean Field Theory**

```
clear all
close all
format long
%load data1D
load Mix2Data
n=size(X,2);
Eta=0.001;
% C=0.1;      %for 1D Gaussian
C=2;          %for 1D Mix
Epsilon=0.0001;
% X=X/3;
% SigmaKernel=0.45;     %for 1D Gaussian
SigmaKernel=1.5;     %for 1D mix
for i=1:n
  for j=1:n
    K(i,j)=Kernel( X(:,i),X(:,j),SigmaKernel,'distribution');
%     K(i,j)=Kernel( X(:,i),X(:,j),SigmaKernel);
  end
  Sigma(i)=K(i,i);
end
```

```
M=10;
Residual=0.05;
Tolerance=0.0005;
while (abs(Residual)>Tolerance)
  Wold=W;
   for i=1:M
     Y=(K*W')';      %To make y as a row
%      Y=Y./maxT;
     Yi=Y-Sigma.*W;
%      Yi=Yi./maxT;

     expPos=0.5*exp(0.5*C*(2*Yi-2*T+2*Epsilon+C*Sigma));
     erfPos=1-erf( (Yi-T+Epsilon+C*Sigma)./(sqrt(2*Sigma)) );
     expNeg=0.5*exp(0.5*C*(-2*Yi+2*T+2*Epsilon+C*Sigma));
     erfNeg=1-erf( (-Yi+T+Epsilon+C*Sigma)./(sqrt(2*Sigma)) );
     erfG=0.5*erf((T-Yi+Epsilon)./(sqrt(2*Sigma)))-0.5*erf((T-Yi-Epsilon)./(sqrt(2*Sigma)));

     F=C*expPos.*(erfPos)-C*expNeg.*(erfNeg);
     G=erfG+expPos.*erfPos+expNeg.*erfNeg;
     W=W+Eta*(F./G-W);
  end
  Y=(K*W')';      %To make y as a row
  Yi=Y-Sigma.*W;
  IG=0.5*erf((T-Yi+Epsilon)./(sqrt(2*Sigma)))-0.5*erf((T-Yi-Epsilon)./(sqrt(2*Sigma)));
  numDW=W.*Yi+C.*C.*Sigma.*IG;
  denDW=Sigma.*G;
  DW=C*C-W.*W-numDW/denDW;
  SIGMAi=-Sigma-1./DW;
  SIGMA=diag(SIGMAi);
  denSigma=diag(inv(SIGMA+K));
  Sigma=abs((1./denSigma)'-SIGMAi);
  Residual=max(abs(W-Wold))/n
end
Y=(K*W')';      %To make y as a row

% X1=[min(X)-3:0.1:max(X)+3];      %For 1D Gaussian
X1=[min(X)-5:0.1:max(X)+5];
for i=1:length(X1)
  Tempfx=0;
  for j=1:n
    Tempfx=Tempfx+W(j)*Kernel( X1(:,i),X(:,j),SigmaKernel,'density');
  end
  fx(i)=Tempfx;
  Ref(i)=0.4*Normal1D(-1,9,X1(i))+0.6*Normal1D(7,4,X1(i));      %for 1D mix
%  Ref(i)=Normal1D(0,1,X1(i));      %for 1D Gaussian
end

KBD=sum((max(Ref)*fx/max(fx)).*log((max(Ref)*fx/max(fx))./Ref))      %Kullback-Leibler distance
figure
plot(X1,Ref,'k');
hold on
plot(X1,max(Ref)*fx/max(fx),'.k')
legend('True', 'Estimated')
```